



PCT/GB 2004 / 0 0 4 6 8 7



INVESTOR IN PEOPLE

BEST AVAILABLE COPY

PRIORITY DOCUMENT
SUBMITTED OR TRANSMITTED IN
COMPLIANCE WITH
RULE 17.1(a) OR (b)

The Patent Office
Concept House
Cardiff Road
Newport
South Wales
NP10 8QQ

REC'D 07 DEC 2004

WIPO

PCT

I, the undersigned, being an officer duly authorised in accordance with Section 74(1) and (4) of the Deregulation & Contracting Out Act 1994, to sign and issue certificates on behalf of the Comptroller-General, hereby certify that annexed hereto is a true copy of the documents as originally filed in connection with the patent application identified therein.

I also certify that the attached copy of the request for grant of a Patent (Form 1/77) bears an amendment, effected by this office, following a request by the applicant and agreed to by the Comptroller-General.

In accordance with the Patents (Companies Re-registration) Rules 1982, if a company named in this certificate and any accompanying documents has re-registered under the Companies Act 1980 with the same name as that with which it was registered immediately before re-registration save for the substitution as, or inclusion as, the last part of the name of the words "public limited company" or their equivalents in Welsh, references to the name of the company in this certificate and any accompanying documents shall be treated as references to the name with which it is so re-registered.

In accordance with the rules, the words "public limited company" may be replaced by p.l.c., p.l.c. or PLC.

Re-registration under the Companies Act does not constitute a new legal entity but merely subjects the company to certain additional company law rules.

Signed

Dated 16 November 2004

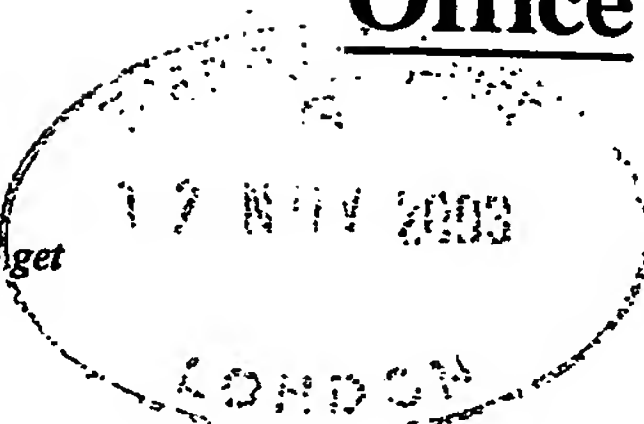
BEST AVAILABLE COPY

Patents Act 1977
(Rule 16)

Request for grant of a patent

(See the notes on the back of this form. You can also get an explanatory leaflet from the Patent Office to help you fill in this form)

The Patent Office
Cardiff Road
Newport
Gwent NP10 8QQ



1. Your reference

12 NOV 2003

A30444

2. Patent application number
(The Patent Office will fill in this part)

0326375.3

13NOV03 F851425-1 003052
P01/7700 0.00-0326375.3

3. Full name, address and postcode of the or of each applicant (*underline all surnames*)

BRITISH TELECOMMUNICATIONS public limited company
81 NEWGATE STREET
LONDON, EC1A 7AJ, England
Registered in England: 1800000

Patents ADP number (*if you know it*)

1867002 ✓

If the applicant is a corporate body, give the country/state of its incorporation

UNITED KINGDOM

4. Title of the invention

OBJECT TRACKING WITHIN VIDEO IMAGES

5. Name of your agent (*if you have one*)

"Address for Service" in the United Kingdom to which all correspondence should be sent (*including the postcode*)

BT Group Legal
Intellectual Property Department
BT Centre
81 Newgate Street
LONDON EC1A 7AJ
SB 26/104

Patents ADP number (*if you know it*)

1867001

08591919001

6. If you are declaring priority from one or more earlier patent applications, give the country and the date of filing of the or of each of these earlier applications and (*if you know it*) the or each application number

Country

Priority application number
(*if you know it*)

Date of filing
(*day / month / year*)

7. If this application is divided or otherwise derived from an earlier UK application, give the number and the filing date of the earlier application

Number of earlier application

Date of filing
(*day/month/year*)

8. Is a statement of inventorship and of right to grant of a patent required in support of this request? (*Answer 'Yes' if:*

- a) any applicant named in part 3 is not an inventor, or
- b) there is an inventor who is not named as an applicant, or
- c) any named applicant is a corporate body.

YES

(See note (d))

Enter the number of sheets for any of the following items you are filing with this form.
Do not count copies of the same document

Continuation sheets of this form -

Description - 14

Claim(s) - 04

Abstract - 01

Drawing(s) - 05 + 5 *ts*

10. If you are also filing any of the following, state how many against each item

Priority Documents

Translations of priority documents

Statement of inventorship and right to grant of a patent (Patents Form 7/77)

Request for preliminary examination and search (Patents Form 9/77) YES

Request for substantive examination (Patents Form 10/77)

Any other documents (please specify)

11. I/We request the grant of a patent on the basis of this application.
Signature(s) Date:
N. Wallin 12 November 2003
WALLIN, Nicholas James, Authorised Signatory

12. Name and daytime telephone number of person to contact in the United Kingdom Rohini RANJITKUMAR 020 7492 8456

Warning

After an application for a patent has been filed, the Comptroller of the Patent Office will consider whether publication or communication of the invention should be prohibited or restricted under Section 22 of the Patents Act 1977. You will be informed if it is necessary to prohibit or restrict your invention in this way. Furthermore, if you live in the United Kingdom, Section 23 of the Patents Act 1977 stops you from applying for a patent abroad without first getting written permission from the Patent Office unless an application has been filed at least 6 weeks beforehand in the United Kingdom for a patent for the same invention and either no direction prohibiting publication or communication has been given, or any such direction has been revoked.

Notes

- If you need help to fill in this form or you have any questions, please contact the Patent Office on 0645 500505.
- Write your answers in capital letters using black ink or you may type them.
- If there is not enough space for all the relevant details on any part of this form, please continue on a separate sheet of paper and write "see continuation sheet" in the relevant part(s). Any continuation sheet should be attached to this form.
- If you have answered 'Yes' Patents Form 7/77 will need to be filed.
- Once you have filled in the form you must remember to sign and date it.
- For details of the fee and ways to pay please contact the Patent Office.

Object Tracking Within Video Images

Technical Field

This invention relates to a method and system for tracking objects detected within
5 video images from frame to frame.

Background to the Invention

Automated video tracking applications are known in the art. Generally, such applications receive video frames as input, and act to detect objects of interest within the
10 image, such as moving objects or the like, frequently using background subtraction techniques. Having detected an object within a single input frame, such applications further act to track detected objects from frame to frame, using characteristic features of the detected objects. By detecting objects in future input frames, and determining the characteristic features of the detected objects, matching of the future detected objects
15 with previously detected objects to produce a track is possible, by matching the determined characteristic features. An example prior art tracking application representative of the above is described within Zhou Q. *et al.* "Tracking and Classifying Moving Objects from Video", *Procs 2nd IEEE Int. Workshop on PETS*, Kauai, Hawaii, USA, 2001.

20 However, matching using characteristic features poses some problems, as some features are more persistent for an object while others may be more susceptible to noise. Also, different features normally assume values in different ranges with different variances. A Euclidean distance matching measure does not account for these factors as it will allow dimensions with larger scales and variances to dominate the distance
25 measure.

Summary of the Invention

The present invention addresses the above by the provision of an object tracking method and system for tracking objects in video frames which takes into account the
30 scaling and variance of each matching feature. This provides for some latitude in the choice of matching feature, whilst ensuring that as many matching features as possible can be used to determine matches between objects, thus giving increased accuracy in the matching thus determined.

In view of the above, from a first aspect the present invention provides a method
35 for tracking objects in a sequence of video images, comprising the steps of:

storing one or more object models relating to objects detected in previous video images of the sequence, the object models comprising values of characteristic features of the detected objects and variances of those values;

receiving a further video image of the sequence to be processed;

5 detecting one or more objects in the received video image;

determining characteristic features of the detected objects;

calculating a distance measure between each detected object and each object model on the basis of the respective characteristic features using a distance function which takes into account at least the variance of the characteristic features;

10 matching the detected objects to the object models on the basis of the calculated distance measures; and

updating the object models using the characteristic features of the respective detected objects matched thereto so as to provide a track of the objects.

15 The use of the distance function which takes into account the variance of the characteristic features compensates for the larger scales and variances of some of the matching characteristic features when compared to others, and hence provides a degree of flexibility in the choice of features, as well as the ability to use as many different matching features as are available to perform a match.

20 In a preferred embodiment the distance measure is a scaled Euclidean distance. This provides the advantage that high-dimensional data can be processed by a computationally inexpensive process, suitable for real-time operation. Preferably the distance function is of the form:-

$$D(l, k) = \sqrt{\sum_{i=1}^N \frac{(x_{li} - y_{ki})^2}{\sigma_{li}^2}}$$

25 for object model l and detected object k , and where the index i runs through all the N features of an object model, and σ_{li}^2 is the corresponding component of the variance of each feature.

In an alternative embodiment the distance measure is the Mahalanobis distance, which takes into account not only the scaling and variance of a feature but also the

values from the object models. By using prediction to predict the values of the characteristic features for each object model for use with the present incoming frame the accuracy of matching of object model to detected object can be increased.

5 In a preferred embodiment, if an object model is not matched to a detected object then the variances of the characteristic feature values of that object are increased. This provides the advantage that it assists the tracker in recovering lost objects that may undergo sudden or unexpected movements.

10 Preferably, if an object model is not matched to a detected object in the received image then the updating step comprises updating the characteristic feature values with an average of each respective value found for the same object over a predetermined number of previous images. This provides for compensation in the case of prediction errors, by changing the prediction model to facilitate re-acquiring the object.

15 Moreover, preferably if an object model is not matched to a detected object in the received image then a test is performed to determine if the object is overlapped with another object, and the object is considered as occluded if an overlap is detected. This provides some flexibility in the track of an object, in that instead of the routine which would ultimately lead to the object being confirmed lost being commenced, if the object is occluded then the tracking technique recognises this as such, and does not immediately remove the object track.

20 Furthermore, the method preferably further comprises counting the number of consecutive video images for which each object is tracked, and outputting a tracking signal indicating that tracking has occurred if an object is tracked for a predetermined number of consecutive frames. This allows short momentary object movements to be discounted.

25 Additionally, if an object model is not matched to a detected object in the received image then preferably a count of the number of consecutive frames for which the object model is not matched is incremented, the method further comprising deleting the object model if the count exceeds a predetermined number. This allows for stationary objects which have become merged with the background and objects that have left the field of view to be discounted by pruning the stored object models relating to such objects, thus
30 maintaining computational efficiency of the technique, and contributing to real-time capability.

Finally, if a detected object is not matched to an object model then preferably a new object model is stored corresponding to the detected object. This allows for new

objects to enter the field of view of the image capture device and to be subsequently tracked.

From a second aspect the present invention also provides a system for tracking objects in a sequence of video images, comprising:-

5 storage means for storing one or more object models relating to objects detected in previous video images of the sequence, the object models comprising values of characteristic features of the detected objects and variances of those values;

means for receiving a further video image of the sequence to be processed; and processing means arranged in use to:-

10 detect one or more objects in the received video image;
determine characteristic features of the detected objects;
calculate a distance measure between each detected object and each object model on the basis of the respective characteristic features using a distance function which takes into account at least the variance of the characteristic features;

15 match the detected objects to the object models on the basis of the calculated distance measures; and

update the stored object models using the characteristic features of the respective detected objects matched thereto.

20 Within the second aspect the same advantages, and same further features and advantages are obtained as previously described in respect of the first aspect.

From a third aspect the present invention also provides a computer program or suite of programs arranged such that when executed on a computer system the program or suite of programs causes the computer system to perform the method of the first aspect. Moreover, from a further aspect there is also provided a computer readable
25 storage medium storing a computer program or suite of programs according to the third aspect. The computer readable storage medium may be any suitable data storage device or medium known in the art, such as, as a non-limiting example, any of a magnetic disk, DVD, solid state memory, optical disc, magneto-optical disc, or the like.

30 Brief Description of the Drawings

FIG. 1 is a block diagram of a system for tracking objects in a sequence of video images, comprising:-
a video capture device 10 for capturing a sequence of video images;
a storage unit 20 for storing one or more object models relating to objects detected in previous video images of the sequence;
a processing unit 30 for receiving a further video image of the sequence to be processed; and
a display unit 40 for displaying the results of the processing.

Figure 2 (a) and (b) are a flow diagram illustrating the operation of the tracking method and system of the embodiment of the invention;

Figure 3 is a drawing illustrating the concept of object templates being matched to detected object blobs used in the embodiment of the invention;

5 Figure 4 is a frame of an video sequence showing the tracking performed by the embodiment of the invention; and

Figure 5 is a later frame of the video sequence including the frame of Figure 4, again illustrating the tracking of objects performed by the invention.

10 Description of an Embodiment

An embodiment of the present invention will now be described with respect to the figures, and an example of the operation of the embodiment given.

Figure 1 illustrates an example system architecture which provides the embodiment of the invention. More particularly, as the present invention generally relates
15 to an image processing technique for tracking objects within input images, the invention is primarily embodied as software to be run on a computer. Therefore, the system architecture of the present invention comprises a general purpose computer 16, as is well known in the art. The computer 16 is provided with a display 20 on which output images generated by the computer may be displayed to a user, and is further provided with
20 various user input devices 18, such as keyboards, mice, or the like. The general purpose computer 16 is also provided with a data storage medium 22 such as a hard disk, memory, optical disk, or the like, upon which is stored programs, and data generated by the embodiment of the invention. An output interface 40 is further provided by the computer 16, from which tracking data relating to objects tracked within the images by the
25 computer may be output to other devices which may make use of such data.

On the data storage medium 22 is stored data 24 corresponding to stored object models (templates), data 28 corresponding to an input image, and data 30 corresponding to working data such as image data, results of calculations, and other data structures or variables or the like used as intermediate storage during the operation of the invention.
30 Additionally stored on the data storage medium 22 is executable program code in the form of programs such as the control program 28, the feature extraction program 32, the matching distance calculation program 36, the object detection program 26, the object models updating program 34, and the predictive filter program 38. The operation of each of these programs will be described in turn later.

In order to facilitate operation of the embodiment, the computer 16 is arranged to receive images from an image capture device 12, such as a camera or the like. The image capture device 12 may be connected directly to the computer 16, or alternatively may be logically connected to the computer 16 via a network 14 such as the Internet. The image capture device 12 is arranged to provide sequential video images of a scene in which objects are to be detected and tracked, the video images being composed of picture elements (pixels) which take particular values so as to have particular luminance and chrominance characteristics. The colour model used for the pixels output from the image capture device 12 may be any known in the art e.g. RGB, YUV, etc.

In operation, the general purpose computer 16 receives images from the image capture device 12 via the network, or directly, and runs the various programs stored on the data storage medium 22 under the general control of the control program 28 so as to process the received input image in order to track objects therein. A more detailed description of the operation of the embodiment will now be undertaken with respect to Figures 2 and 3.

With reference to Figure 2, at step 2.2 a new video image is received from the image capture device 12, forming part of a video sequence being received from the device. For the sake of this description, we assume that previous images have been received, and that objects have previously been detected and tracked therein; a brief description of the start-up operation when the first images of a sequence are received is given later.

Following step 2.2, the first processing to be performed is that objects of interest (principally moving objects) need to be detected within the input image, a process generally known as "segmentation". Any segmentation procedure already known in the art may be used, such as those described by McKenna et al. in "Tracking Groups of People", *Computer Vision and Image Understanding*, 80, 42-56, 2000 or by Horprasert et al. in "A Statistical Approach for Real-time Robust Background Subtraction and Shadow Detection" *IEEE ICCV'99 FRAME_RATE workshop*, the contents of either document necessary for understanding the present invention being incorporated herein by reference. Alternatively,

and operation however in object detection techniques as described in the present invention, the present invention is not limited to the use of the above mentioned techniques, but may be implemented using any suitable object detection technique known in the art.

throughout their movements within the scene by means of temporal templates (object models). The contents of the object templates is discussed next.

More particularly, and as shown in Figure 3, each object of interest that has been previously tracked within the scene represented by the input images is modelled by a temporal template of persistent characteristic features. In the present embodiment, a set of five significant features are used describing the velocity, shape, and colour of each object / candidate blob, namely:

- the velocity $\mathbf{v} = (v_x, v_y)$ at its centroid (p_x, p_y) ;
- the size, or number of pixels, contained (s);
- the ratio of the major-axis vs. minor-axis of the ellipse (r) that best fits the blob - this ratio of the ellipse better describes an object than the aspect ratio of its bounding box;
- the orientation of the major-axis of the ellipse (θ); and
- the dominant colour representation (c_p), using the principal eigenvector of the aggregated pixels' colour covariance matrix of the blob.

Therefore, at any time t , we have, for each tracked object / centred at (p_{tx}, p_{ty}) , a template of features:

$$M_l(t) = (v_l, s_l, r_l, \theta_l, 1_l(c_p))$$

These object models (or templates) are stored in the data storage medium 22 in the object models area 24.

There are two aspects of the above that need special clarification as follows:

- a) Prior to matching the template l with a candidate blob k in frame $t+1$, centred at (p'_{tx}, p'_{ty}) with a feature vector $B_k(t+1) = (v'_k, s'_k, r'_k, \theta'_k, d_k(c'_p))$, Kalman filters are used to update the template by predicting, respectively, its new velocity, size, aspect ratio, orientation in $\hat{M}_l(t+1)$. The velocity of the candidate blob k is calculated as,

$$\mathbf{v}'_k = (p'_{tx}, p'_{ty})^T - (p_{tx}, p_{ty})^T.$$

- b) Instead of c_p , we use $1_l(c_p)$, or the value of 1.0, to denote the dominant colour of the template, and $d_k(c'_p)$ defined in Eq. (1), to represent the colour similarity between the template l and candidate blob k :

$$d_k(c'_p) = \frac{\mathbf{c}_p \cdot \mathbf{c}'_p}{\|\mathbf{c}_p\| \cdot \|\mathbf{c}'_p\|} \quad (1)$$

It is only after a match (described later) is found that the template's dominant colour is replaced with that of the matched candidate.

Additionally, the stored object models also include the mean $\overline{M}_i(t)$ and variance $V_i(t)$ vector; these values are updated whenever a candidate blob k in frame $t+1$ is found to match with the template. The mean $\overline{M}_i(t)$ and variance $V_i(t)$ vector are computed using the latest corresponding L blobs that the object has matched, or a temporal window of L frames (e.g., $L=50$). With regard to individual Kalman filters $KF_i(t)$, they are updated only by feeding with the corresponding feature value of a matched blob.

In view of the above, i.e. that the tracked objects are tracked by means of stored object templates comprising specific matching features and the mean, variance, and predicted values of each, in order to match the stored object models with the detected objects, at step 2.6 the feature extraction program 32 acts to detect the object matching characteristic features, as outlined above i.e. for a candidate blob k in frame $t+1$, centred at (p'_{kx}, p'_{ky}) the feature vector $B_k(t+1) = (v'_k, s'_k, r'_k, \theta'_k, d_k(c'_p))$ is detected. Note that a respective feature vector is determined for every detected object in the present input frame $t+1$.

Having calculated the detected objects' feature vectors, it is then possible to begin matching the detected objects to the tracked objects represented by the stored object templates. Therefore, at step 2.8 the matching distance calculation program 36 is launched, which commences a FOR processing loop which generates an ordered list of matching distances for every stored object template with respect to every detected object in the input image. More particularly, at the first iteration of step 2.8 the first stored object template is selected, and its feature vector retrieved. Then, at step 2.10 a second nested FOR processing loop is commenced, which acts to step through the feature vectors of every detected object, processing each set in accordance with step 2.12. At step 2.12 a matching distance value is calculated between the present object template and the present detected object being processed, by comparing the respective matching features to determine a matching distance therebetween. Further details of the matching function applied at step 2.12 are given next.

As described above, the template for each object being tracked has a set of associated Kalman filters that predict the expected values for each feature component for the

One way to tackle this problem is to use the Mahalanobis distance metric, which takes into account not only the scaling and variance of a feature, but also the variation of other features based on the covariance matrix. Thus, if there are correlated features, their contribution is weighted appropriately. In an alternative embodiment such a distance
5 metric may be employed.

However, with high-dimensional data, the covariance matrix can become non-invertible. Furthermore, matrix inversion is a computationally expensive process, not suitable for real-time operation. So, in the present embodiment a scaled Euclidean distance, shown in Eq. (2), between the template l and a candidate blob k is adopted,
10 assuming a diagonal co-variance matrix. For a heterogeneous data set, this is a reasonable distance definition.

$$D(l, k) = \sqrt{\sum_{i=1}^N \frac{(x_{li} - y_{ki})^2}{\sigma_{li}^2}} \quad (2)$$

where the index i runs through all the features of the template, and σ_{li}^2 is the corresponding component of the variance vector $V_l(t)$. Note especially that for the colour
15 component, $x_{li} = 1.0$ is assumed for the object l , and $y_{ki} = d_k(c'_p)$ for the candidate blob k .

Following step 2.12, at step 2.14 an evaluation is performed to determine whether all of the detected objects have been matched against the present object template being processed i.e. whether the inner FOR loop has finished. If not, then the next detected object is selected, and the inner FOR loop repeated. If so, then processing
20 proceeds to S.2.16.

At step 2.16 the present state of processing is that a list of matching distances matching every detected object against the stored object template being presently processed has been obtained, but this list is not ordered, and neither has it been checked to determine if the distance measure values are reasonable. In view of this, at step 2.16 a
25 threshold is applied to the distance values in the list, and those values which are greater than the threshold are pruned out of the list. A *THR* value of 10 proved to work in practice, but other values should also be effective. Following the thresholding operation, at step 2.18 the resulting thresholded list is ordered by matching distance value, using a standard sort routine.

30 Next, at step 2.20 an evaluation is performed to determine whether all of the stored object templates have been processed i.e. whether the outer FOR loop has finished. If not, then the next object template is selected, and the outer and inner FOR loops repeated. If so, then processing proceeds to S.2.22.

At this stage in the processing, we have stored in the working data area 30 respective ordered lists of matching distances, one for each stored object model. Using these ordered lists it is then possible to match detected objects to the stored object models, and this is performed next.

5 More particularly, within the embodiment at step 2.22 a second FOR processing loop is commenced, which again acts to perform processing steps on each stored object template. In particular, firstly at step 2.24 an evaluation is performed to determine if the object model being presently processed has an available match. A match is made with the detected object which gave the lowest matching distance value in the present object
10 model's ordered. No match is available if, due to the thresholding step carried out previously, there are no matching distance values in the present object model's ordered list.

If the evaluation of step 2.24 returns true i.e, present object l is matched by a candidate blob k in frame $t+1$, i.e. by way of the template prediction $\hat{M}_l(t+1)$, variance
15 vector $V_l(t)$ and $B_k(t+1)$, then processing proceeds to step 2.26 and the updates for the present object model l are performed. In particular the object template for the present object is updated by the object models updating program 34 to obtain $M_l(t+1) = B_k(t+1)$ with $1_l(c_p)$ replaced by $1_k(c'_p)$, as well as the mean and variance $(\bar{M}_l(t+1), V_l(t+1))$. Correspondingly the Kalman filters for the object model are also updated with the values
20 of the matched detected object at step 2.28 using the predictive filter program 38, and the predicted values for the features of the object model for use with the next input frame are determined. Additionally, at step 2.30 a 'TK_counts' counter value representing the number of frames for which the object has been tracked is increased by 1, and an 'MS_counts' counter which may have been set if the track of the object had been
25 temporarily lost in the preceding few frames is set to zero at step 2.32. The FOR loop then ends with an evaluation as to whether all of the stored object templates have been processed, and if so processing proceeds to step 2.56 (described later). If all of the stored object templates have not been processed, then the FOR loop of s.2.22 is recommenced with the next stored object template to be processed.

step 2.36, wherein the *TK_counts* counter for the present object template is evaluated to determine whether it is less than a predetermined value *MIN_SEEN*, which may take a value of 20 or the like. If *TK_counts* is less than *MIN_SEEN* then processing proceeds to step 2.54, wherein the present object template is deleted from the object model store 24.

- 5 Processing then proceeds to step 2.34, shown as a separate step on the diagram, but in reality identical to that described previously above. This use of the *MIN_SEEN* threshold value is to discount momentary object movements and artefact blobs which may be temporarily segmented but which do not in fact correspond to proper objects to be tracked.
- 10 If the evaluation of step 2.36 indicates that the *TK_counts* counter exceeds the *MIN_SEEN* threshold then a test for occlusion is next performed, at step 2.38. In the present embodiment, no use is made of any special heuristics concerning the areas where objects enter/exit into/from the scene. Objects may just appear or disappear in the middle of the image, and, hence, positional rules are not necessary. To handle occlusions,
- 15 therefore, the use of heuristics is essential. As a result within the embodiment every time an object has failed to find a match with a detected object a test on occlusion is carried out at step 2.38. Here, if the present object's bounding box overlaps with some other object's bounding box, as determined by the evaluation at step 2.40, then both objects are marked as 'occluded' at step 2.42. Processing then proceeds to step 2.48, which will be
- 20 described below.

Returning to step 2.40, if the occlusion test indicates that there are no overlapping other templates i.e. the present object is not occluded, then the conclusion is drawn that the tracking of the object has been lost. Therefore, processing proceeds to s.2.48 where an *MS_counts* counter is incremented, to keep a count of the number of

25 input frames for which the tracking of a particular object model has not been successful. At step 2.50 this count is compared with a threshold value *MAX_LOST*, which may take a value such as 5 or the like. If this evaluation indicates that the counter is greater than or equal to the threshold, then the conclusion is drawn that the tracking of the object has been irretrievably lost, and hence processing process to step 2.54, wherein the present

30 object model is deleted, as previously described above.

If, however, the evaluation of step 2.50 indicates that the counter is less than *MAX_LOST* then processing proceeds to step 2.52, wherein the variance values of the object model are adjusted according to Eq. (3):

$$\sigma_i^2(t+1) = (1 + \delta)\sigma_i^2(t) \quad (3)$$

where $\delta = 0.05$ is a good choice. This increase in the variance assists the tracker to recover lost objects that have undergone unexpected or sudden movements.

Following step 2.52, processing proceeds to step 2.44. Note also that step 2.44 can also be reached from step 2.42, where the present object model is marked as being occluded. As an error in the matching can occur simply due to the prediction errors, at step 2.44 the prediction model is changed to facilitate the possible recovery of the lost tracking. Hence within the *MAX_LOST* period, Kalman filters are not used to update the template of features but instead, at step 2.44 for each feature an average of the last 50 correct predictions is used, which states as $M_i(t+1) = M_i(t) + \overline{M}_i(t)$. Moreover, if an object is marked as being occluded then the same update is performed. This is because occluded objects are better tracked using the averaged template predictions, as small erratic movements in the last few frames are then filtered out. Predictions of positions are also constrained within the occlusion blob.

Following step 2.44, processing proceeds to the evaluation of step 2.34, which
15 has already been described.

Once the evaluation of step 2.34 indicates that every object template has been processed in accordance with the processing loop commenced at s.2.22, the present state of processing is that every stored object model will have been either matched with a detected object, marked as occluded, not matched but within the *MAX_LOST* period, or
20 deleted from the object model store 24 (either by virtue of no match having been found within the *MIN_SEEN* period, or by virtue of the *MAX_LOST* period having been exceeded without the object having been re-acquired). However, there may still be detected objects in the image which have not been matched to a stored object model, usually because they are new objects which have just appeared within the image scene
25 for the first time in the present frame (for example, a person walking into the image field of view from the side). In order to account for these unmatched detected objects, new object models must be instantiated and stored in the object model store.

To achieve this, following step 2.34 (once it indicates that every object template has been processed in accordance with the processing loop commenced at s.2.22).

1. The first step in the process is to identify the problem or issue that needs to be addressed. This involves gathering information and understanding the context of the problem.

2. Once the problem is identified, the next step is to define the objectives and goals of the project. This helps to clarify what needs to be achieved and provides a clear direction for the team.

3. The third step is to develop a plan or strategy to address the problem. This involves breaking down the problem into smaller, manageable tasks and determining the resources needed to complete each task.

4. The fourth step is to implement the plan. This involves putting the strategy into action and monitoring progress to ensure that the project is on track.

5. The final step is to evaluate the results of the project. This involves assessing the outcomes against the objectives and goals and identifying any areas for improvement.

model for the detected object, and hence processing proceeds to step 2.62. Step 2.62 determines whether or not all the detected objects have been processed by the FOR loop commenced at step 2.56, and returns the processing to step 2.56 to process the next detected object if not, or ends the FOR if all the detected objects have been processed.

- 5 If the present detected object has not been matched with a stored object model, however, then a new object model must be instantiated and stored at step 2.60, taking the detected object's feature values as it's initial values i.e. for the present detected object k in frame $t+1$, a new object template $M_k(t+1)$ is created from $B_k(t+1)$. The choice of initial variance vector $V_k(t+1)$ for the new object needs some consideration, but suitable values
- 10 can either be copied from very similar objects already in the scene or taken from typical values obtained by prior statistical analysis of correctly tracked objects, as a design option. The new object model is stored in the object model store 24, and hence will be available to be matched against when the next input image is received.

- Following step 2.60 the loop evaluation of step 2.62 is performed as previously
- 15 described, and once all of the detected object have been processed by the loop processing can proceed onto step 2.64. At this stage in the processing all of the stored object models have been matched to detected objects, marked as occluded or lost within the *MAX_LOST* period, or deleted, and all of the detected objects have either been matched to stored object models, or had new object models created in respect thereof. It
- 20 is therefore possible at this point to output tracking data indicating the matches found between detected objects and stored object models, and indicating the position of tracked objects within the image. Therefore, at step 2.64 a tracking output is provided indicating the match found for every stored object template for which the *TK_counts* counter is greater than the *MIN_SEEN* threshold. As mentioned previously, the use of the
- 25 *MIN_SEEN* threshold allows any short momentary object movements to be discounted, and also compensates for artefact temporarily segmented blobs which do not correspond to real objects. Moreover, as we have seen, object models are deleted if the tracking of the object to which they relate is lost (i.e. the object model is not matched) within the *MIN_SEEN* period.

- 30 Within the embodiment the output tracking information is used to manipulate the image to place a visible bounding box around each tracked object in the image, as shown in Figures 4 and 5. Figures 4 and 5 are two frames from a video sequence which are approximately 40 frames temporally separated (Figure 5 being the later frame). Within these images it will be seen that bounding boxes provided with object reference numbers
- 35 have been placed around the tracked objects, and by a comparison of Figure 4 to Figure 5

it will be seen that the objects within the scene are tracked as they move across the scene (indicated here by the bounding boxes around each object having the same reference numbers in each image). Moreover, Figure 5 illustrates the ability of the present embodiment to handle occlusions, as the group of people tracked as object 956 are
5 occluded by the van tracked as object 787, but each object has still been successfully tracked.

As well as simply indicating that an object is being tracked by providing a visual output on the image, the tracking information provided by the embodiment may be employed in further applications, such as object classification applications or the like.
10 Furthermore, the tracking information may be output at the tracking output 40 of the computer 16 (see Figure 1) to other systems which may make use of it. For example the tracking information may be used as input to a device pointing system for controlling a device such as a camera or a weapon to ensure that the device remains pointed at a particular object in an image as the object moves. Other uses of the tracking information
15 will be apparent to those skilled in the art.

Unless the context clearly requires otherwise, throughout the description and the claims, the words "comprise", "comprising" and the like are to be construed in an inclusive as opposed to an exclusive or exhaustive sense; that is to say, in the sense of "including, but not limited to".

20 Moreover, for the avoidance of doubt, where reference has been given to a prior art document or disclosure, whose contents, whether as a whole or in part thereof, are necessary for the understanding of the operation or implementation of any of the embodiments of the present invention by the intended reader, then said contents should be taken as being incorporated herein by said reference thereto.

CLAIMS

1. A method for tracking objects in a sequence of video images, comprising the steps of:
 - 5 storing one or more object models relating to objects detected in previous video images of the sequence, the object models comprising values of characteristic features of the detected objects and variances of those values;
 - receiving a further video image of the sequence to be processed;
 - detecting one or more objects in the received video image;
 - 10 determining characteristic features of the detected objects;
 - calculating a distance measure between each detected object and each object model on the basis of the respective characteristic features using a distance function which takes into account at least the variance of the characteristic features;
 - matching the detected objects to the object models on the basis of the calculated
 - 15 distance measures; and
 - updating the object models using the characteristic features of the respective detected objects matched thereto.
2. A method according to claim 1, wherein the distance measure is a scaled
- 20 Euclidean distance.
3. A method according to claim 2, wherein the distance function is of the form:-

$$D(l, k) = \sqrt{\sum_{i=1}^N \frac{(x_{li} - y_{ki})^2}{\sigma_{li}^2}}$$

for object model l and detected object k , and where the index i runs through all the

- 25 features of an object model, and σ_{li}^2 is the corresponding component of the variance of each feature.
- 4. A method according to claim 1, wherein the distance measure is the Mahalanobis distance.
- 30 5. A method according to any of the preceding claims, and further comprising the step of predicting the values of the characteristic features of the stored object models for

the received frame; wherein the calculating step uses the predicted values of the characteristic features as the feature values from the object models.

6. A method according to any of the preceding claims, wherein if an object model is not matched to a detected object then the variances of the characteristic feature values of that object are increased.

7. A method according to any of the preceding claims, wherein if an object model is not matched to a detected object in the received image then the updating step comprises updating the characteristic feature values with an average of each respective value found for the same object over a predetermined number of previous images.

8. A method according to any of the preceding claims, wherein if an object model is not matched to a detected object in the received image then a test is performed to determine if the object is overlapped with another object, and the object is considered as occluded if an overlap is detected.

9. A method according to any of the preceding claims, further comprising counting the number of consecutive video images for which each object is tracked, and outputting a tracking signal indicating that tracking has occurred if an object is tracked for a predetermined number of consecutive frames.

10. A method according to any of the preceding claims, wherein if an object model is not matched to a detected object in the received image then a count of the number of consecutive frames for which the object model is not matched is incremented, the method further comprising deleting the object model if the count exceeds a predetermined number.

11. A method according to any of the preceding claims, wherein if a detected object is not matched to an object model then a new object model is stored corresponding to the detected object.

13. A computer readable storage medium storing a computer program or at least one of a suite of computer programs according to claim 12.

5 14. A system for tracking objects in a sequence of video images, comprising:-
 storage means for storing one or more object models relating to objects detected
 in previous video images of the sequence, the object models comprising values of
 characteristic features of the detected objects and variances of those values;
 means for receiving a further video image of the sequence to be processed; and
 10 processing means arranged in use to:-
 detect one or more objects in the received video image;
 determine characteristic features of the detected objects;
 calculate a distance measure between each detected object and each
 object model on the basis of the respective characteristic features using a distance
 15 function which takes into account at least the variance of the characteristic features;
 match the detected objects to the object models on the basis of the
 calculated distance measures; and
 update the stored object models using the characteristic features of the
 respective detected objects matched thereto.

20

15. A system according to claim 14, wherein the distance measure is a scaled Euclidean distance.

16. A system according to claim 15, wherein the distance function is of the form:-

25
$$D(l,k) = \sqrt{\sum_{i=1}^N \frac{(x_{li} - y_{ki})^2}{\sigma_{li}^2}}$$

for object model l and detected object k , and where the index i runs through all the features of an object model; and σ_{li}^2 is the corresponding component of the variance of each feature.

30 17. A system according to claim 14, wherein the distance measure is the Mahalanobis distance.

18. A system according to any of claims 14 to 17, and further comprising means for predicting the values of the characteristic features of the stored object models for the received frame; wherein the processing means uses the predicted values of the characteristic features as the feature values from the object models within the distance
5 measure calculation.

19. A system according to any of claims 14 to 18, wherein if an object model is not matched to a detected object then the variances of the characteristic feature values of that object are increased.

10

20. A system according to any of claims 14 to 19, wherein if an object model is not matched to a detected object in the received image then the updating step comprises updating the characteristic feature values with an average of each respective value found for the same object over a predetermined number of previous images.

15

21. A system according to any of claims 14 to 20, wherein if an object model is not matched to a detected object in the received image then a test is performed to determine if the object is overlapped with another object, and the object is considered as occluded if an overlap is detected.

20

22. A system according to any of claims 14 to 21, further comprising means for counting the number of consecutive video images for which each object is tracked, and means for outputting a tracking signal indicating that tracking has occurred if an object is tracked for a predetermined number of consecutive frames.

25

23. A system according to any of claims 14 to 22, wherein if an object model is not matched to a detected object in the received image then a count of the number of consecutive frames for which the object model is not matched is incremented, the system further comprising means for deleting the object model if the count exceeds a
30 predetermined number.

**ABSTRACT****Object Tracking Within Video Images**

This invention provides an object tracking method and system for tracking objects in video frames which takes into account the scaling and variance of each matching feature. This provides for some latitude in the choice of matching feature, whilst ensuring that as many matching features as possible can be used to determine matches between objects, thus giving increased accuracy in the matching thus determined. A parallel matching approach is used, and heuristic rules employed to account for occlusions between objects.

10

Figure (1)

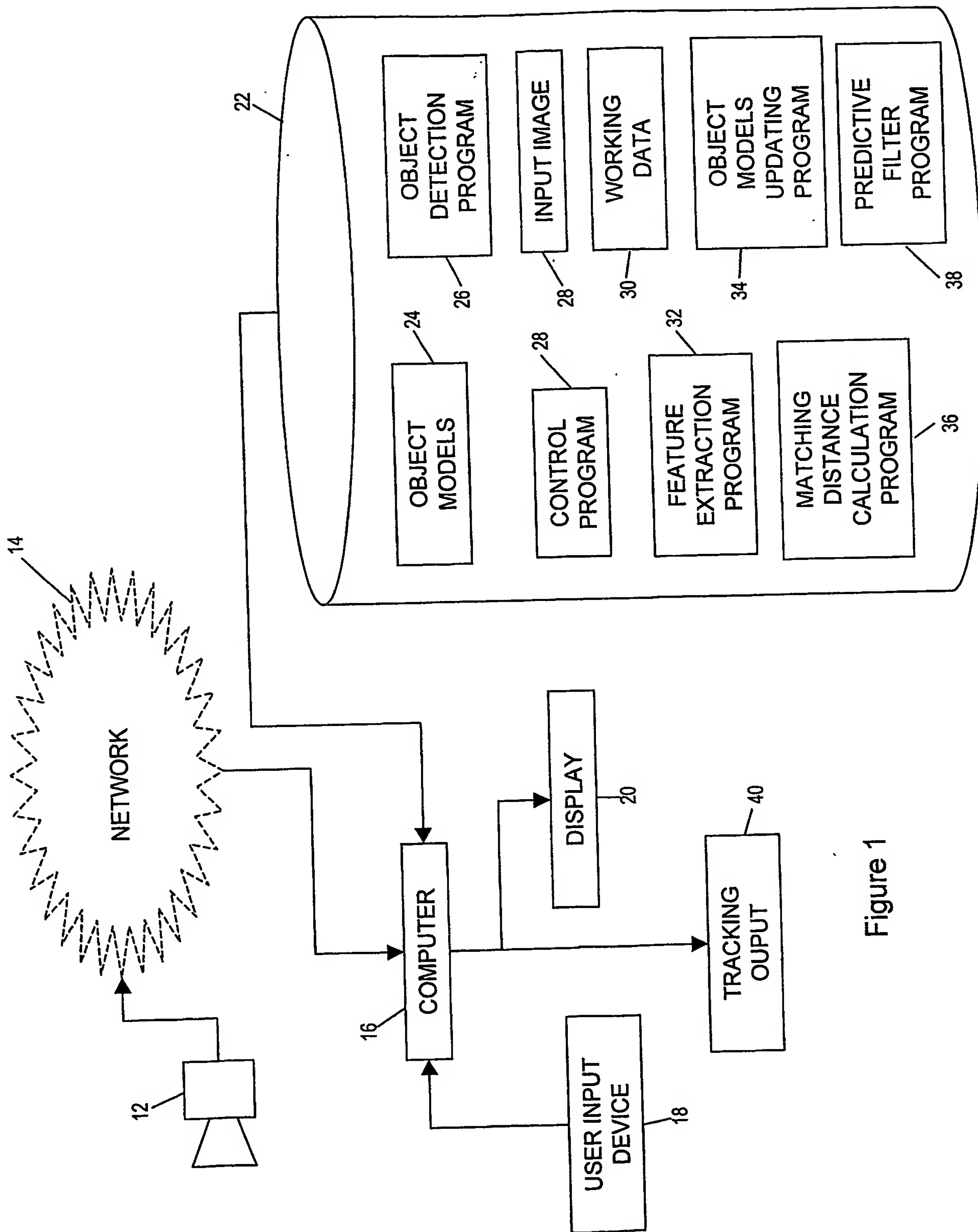
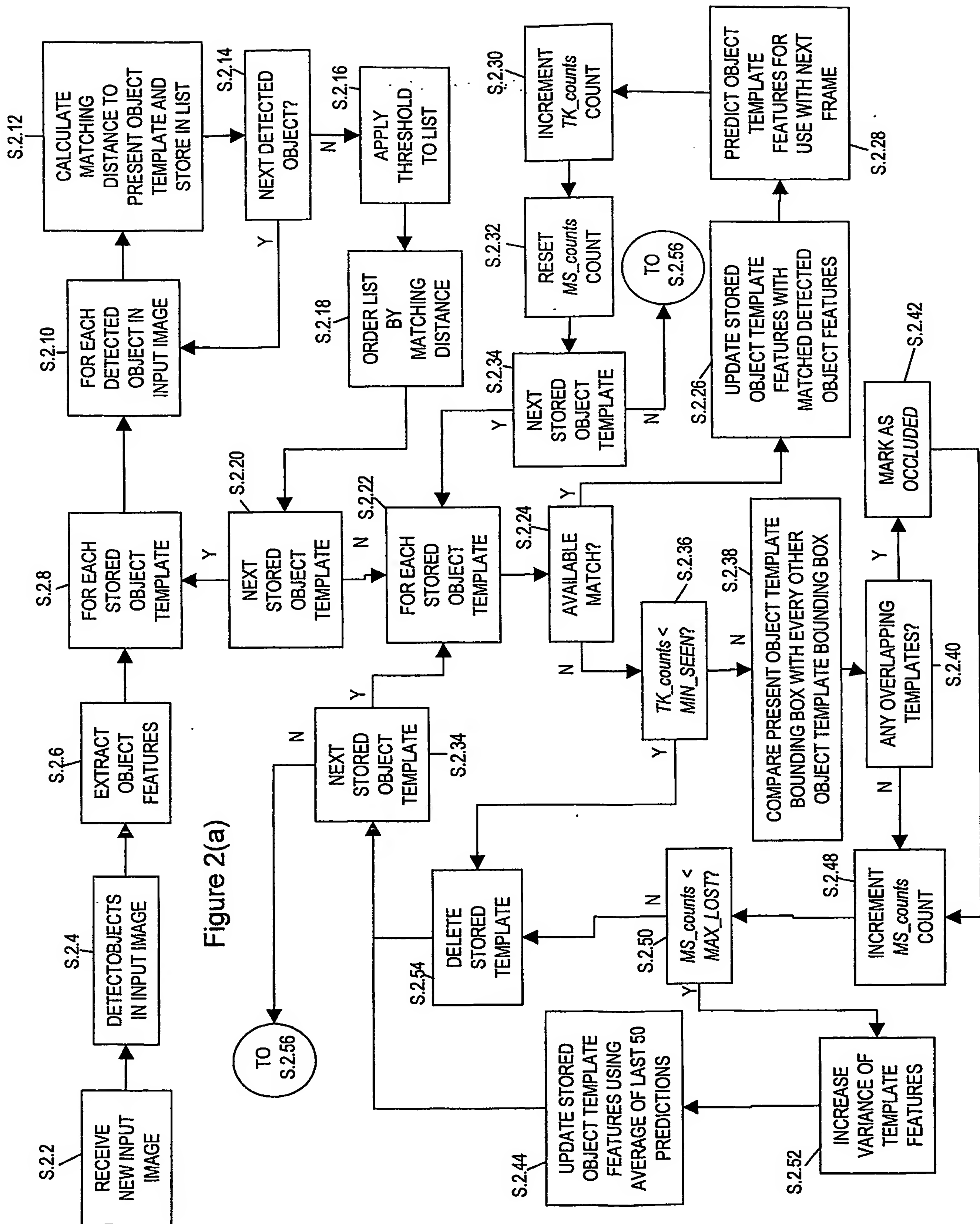


Figure 1



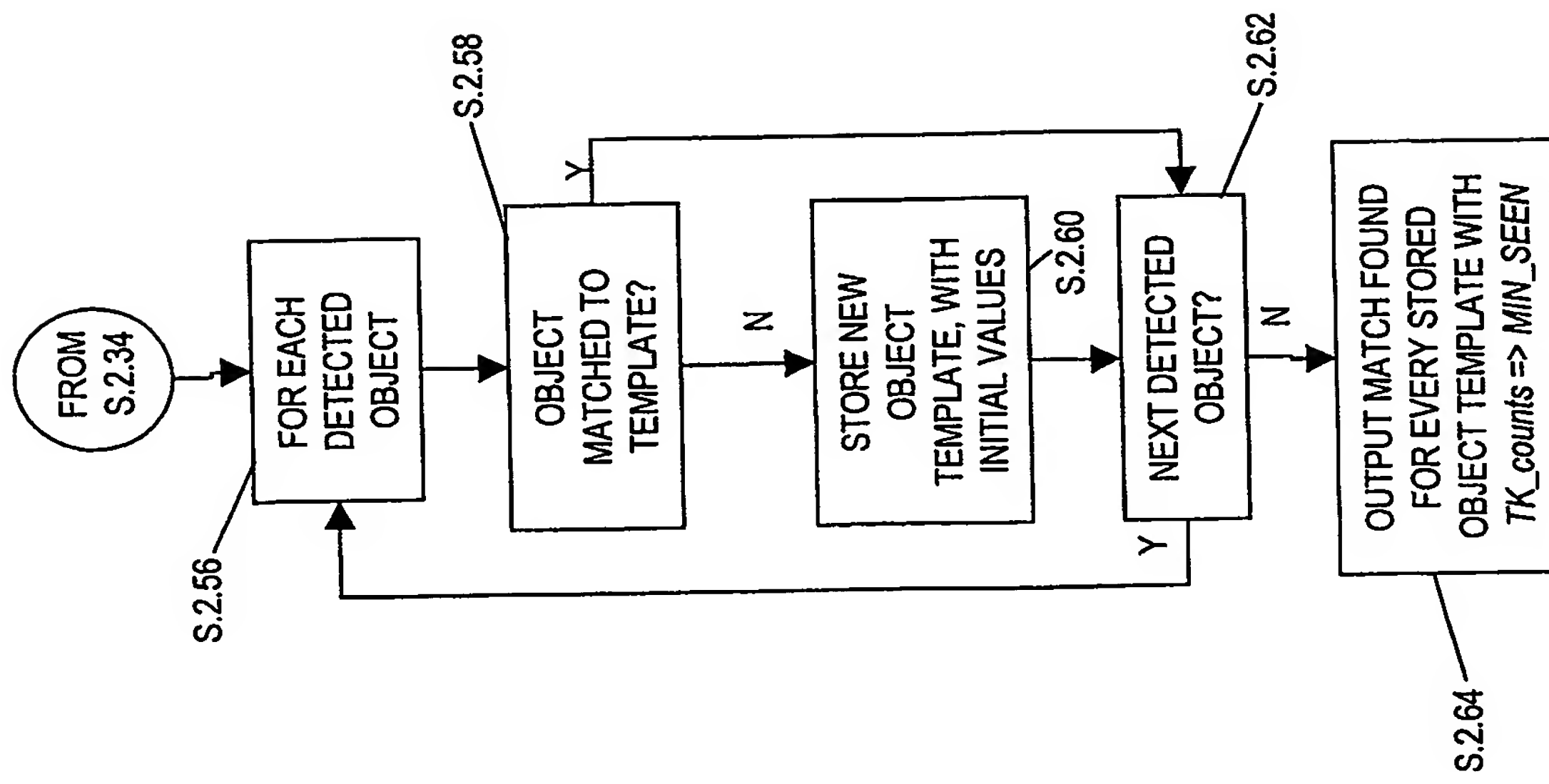


Figure 2(b)

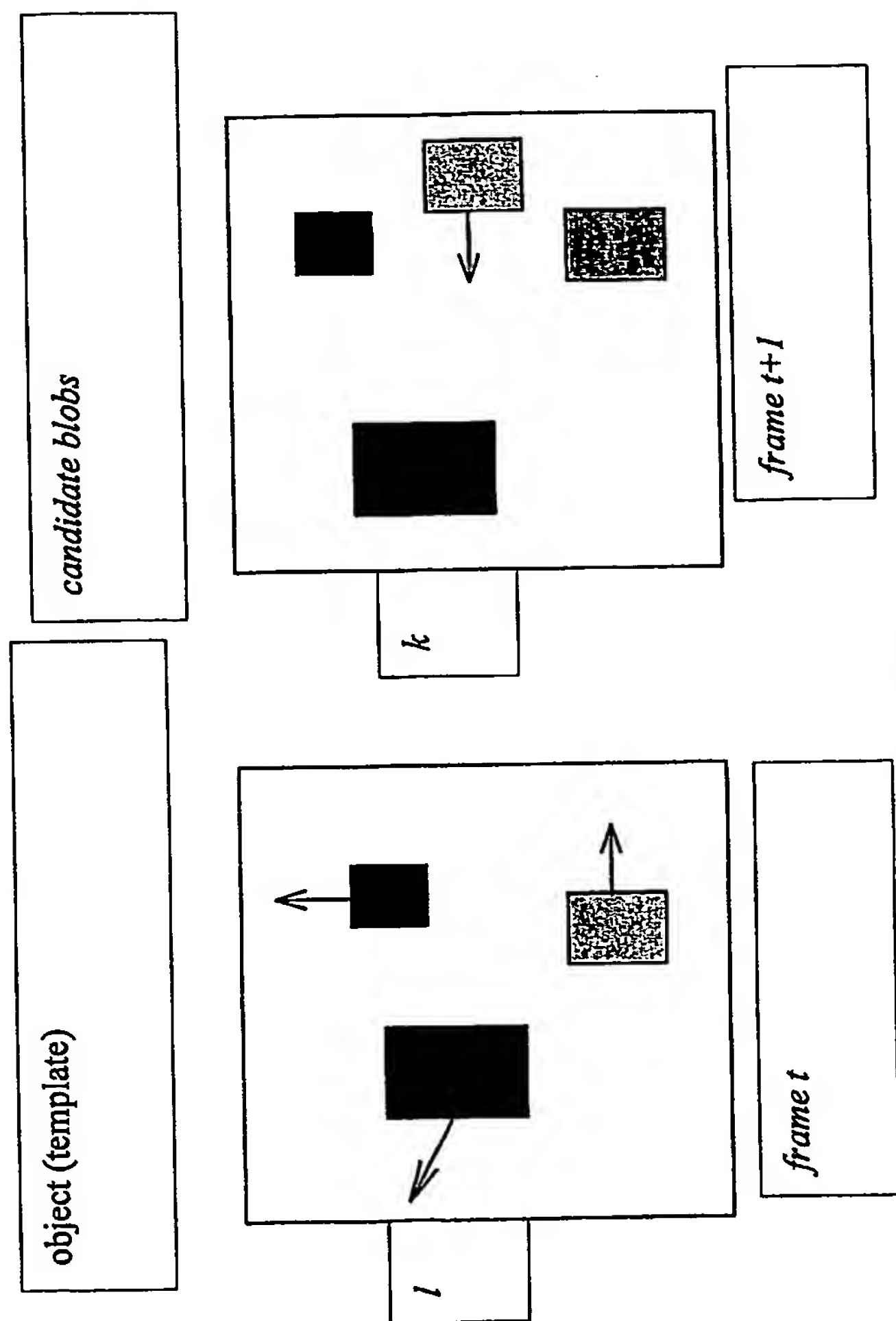


Figure 3

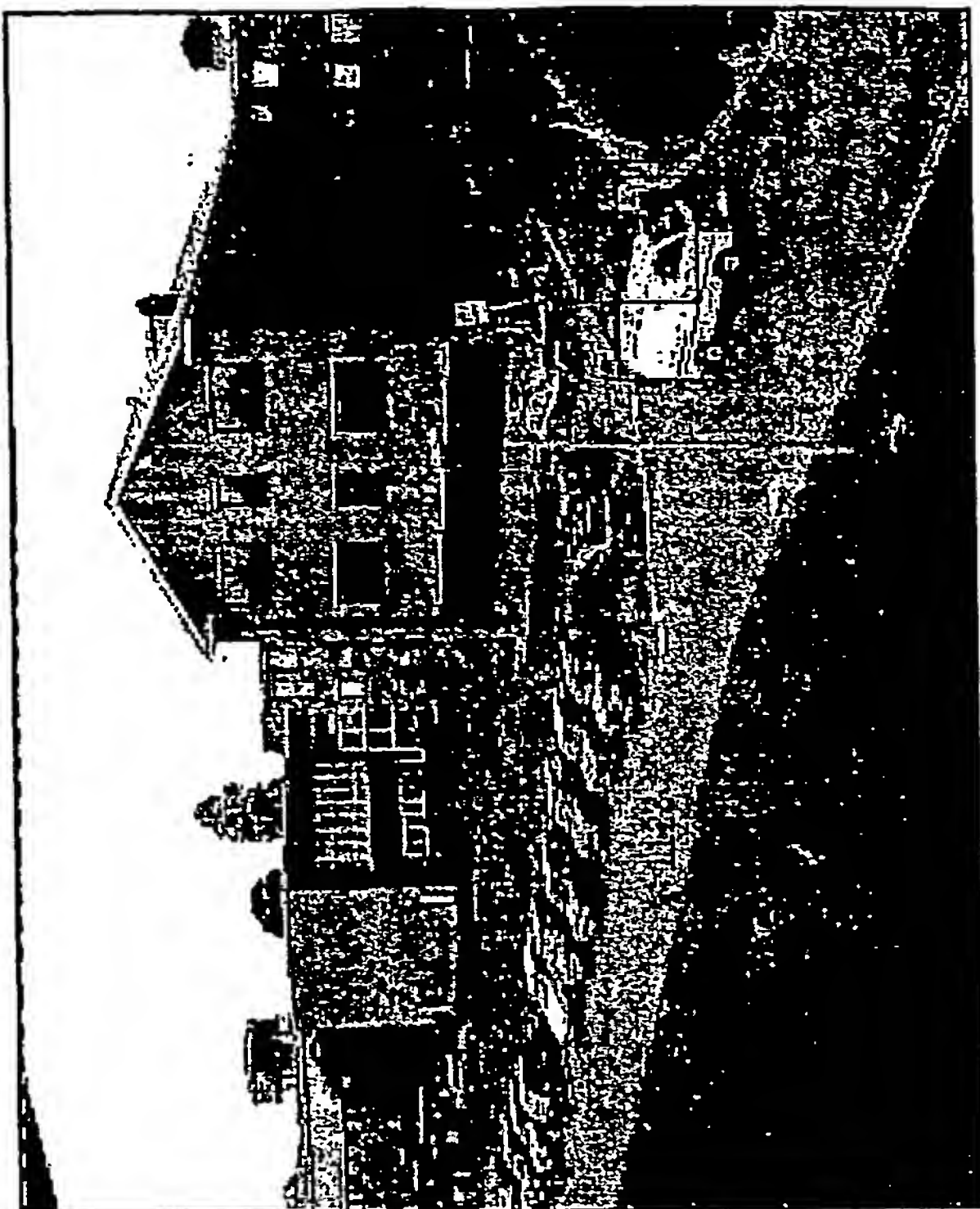


Figure 5

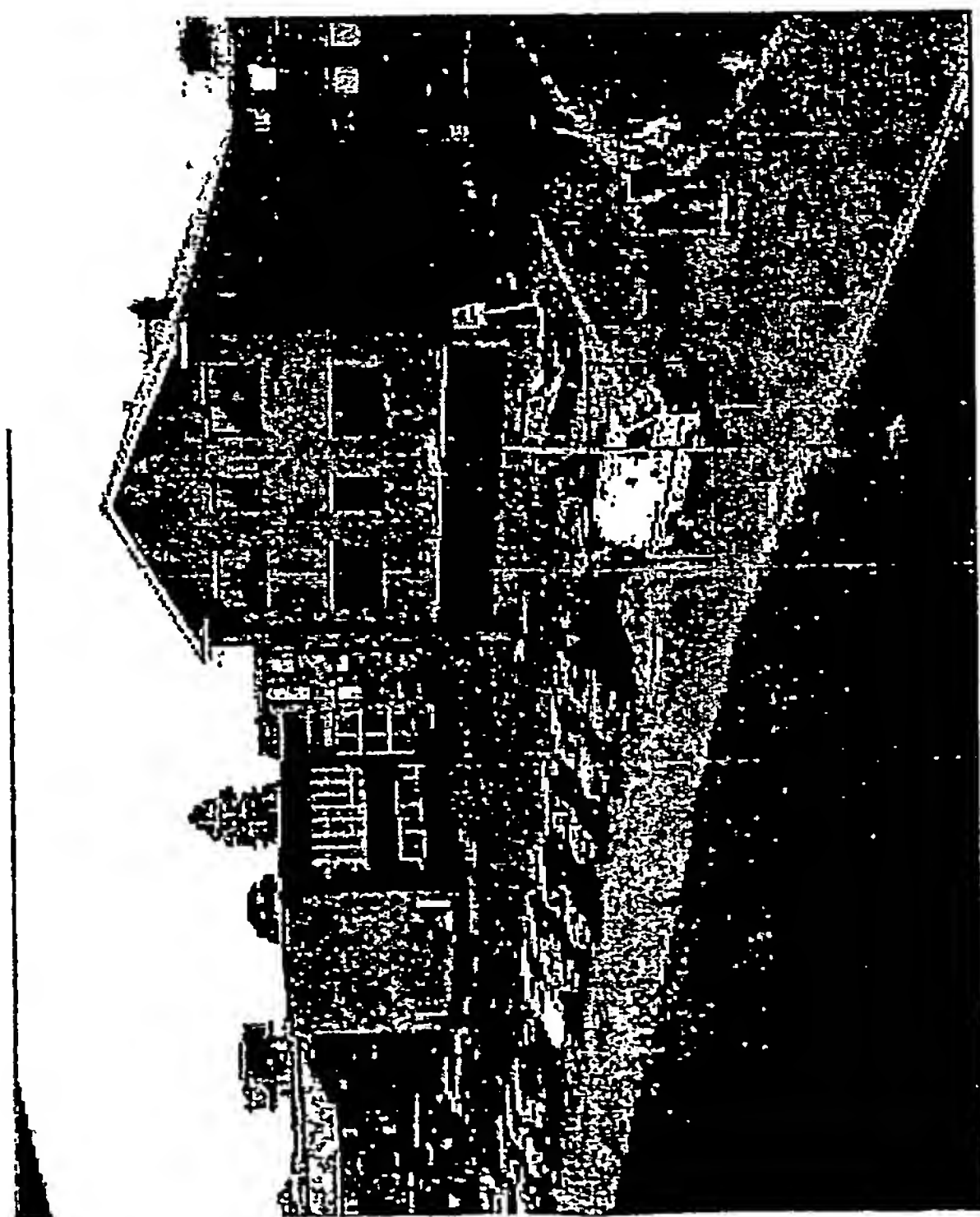


Figure 4

PCT/GS2004/004627

8/11/04.

BT GROUP LEGAL.

**This Page is Inserted by IFW Indexing and Scanning
Operations and is not part of the Official Record**

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ **BLACK BORDERS**
- ☐ **IMAGE CUT OFF AT TOP, BOTTOM OR SIDES**
- ☐ **FADED TEXT OR DRAWING**
- ☐ **BLURRED OR ILLEGIBLE TEXT OR DRAWING**
- ☐ **SKEWED/SLANTED IMAGES**
- ☐ **COLOR OR BLACK AND WHITE PHOTOGRAPHS**
- ☐ **GRAY SCALE DOCUMENTS**
- ☒ **LINES OR MARKS ON ORIGINAL DOCUMENT**
- ☐ **REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY**
- ☐ **OTHER:** _____

IMAGES ARE BEST AVAILABLE COPY.

As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.